MUNI
FI

lasaris LAB OF SOFTWARE ARCHITECTURES AND INFORMATION SYSTEMS

Object Diagram

PB007 Software Engineering I

Lukáš Daubner
daubner@mail.muni.cz

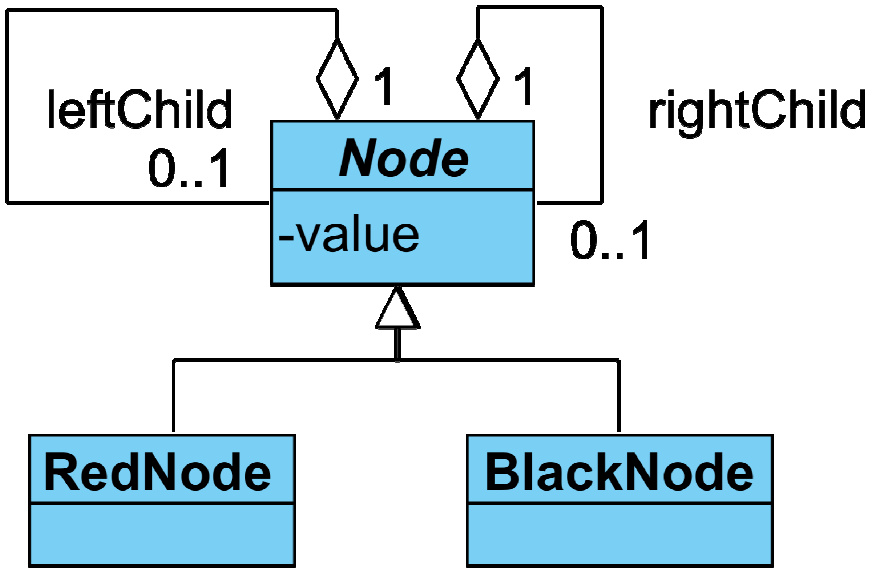
Sequence Diagram

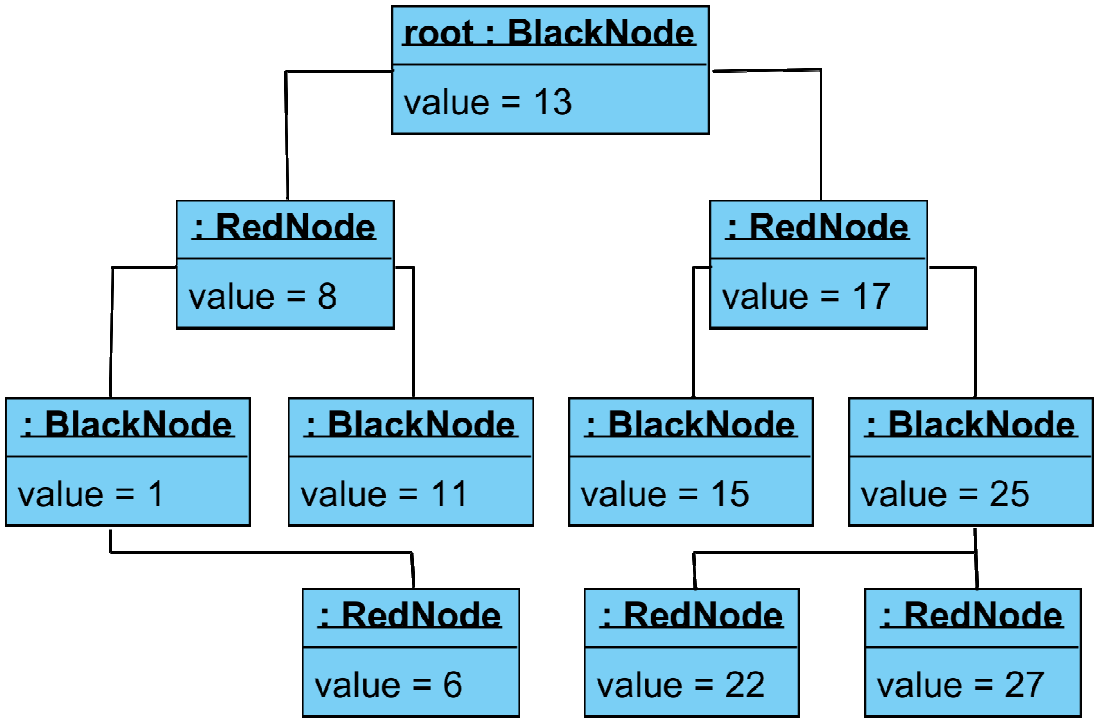
- The name of the diagram should ideally be according to the UC we are modelling
- The actor is not an object and accesses the system usually through service classes (services / managers)
- Objects do not appear out of nothing

Object Diagram

- Instance of Class Diagram
  - Well, duh!
- Represents a **snapshot** of a system state in a given time
  - Objects with values and their relationships in a particular moment
- It is still a static view of the system
  - The dynamics frozen in time

Object Diagram – Example





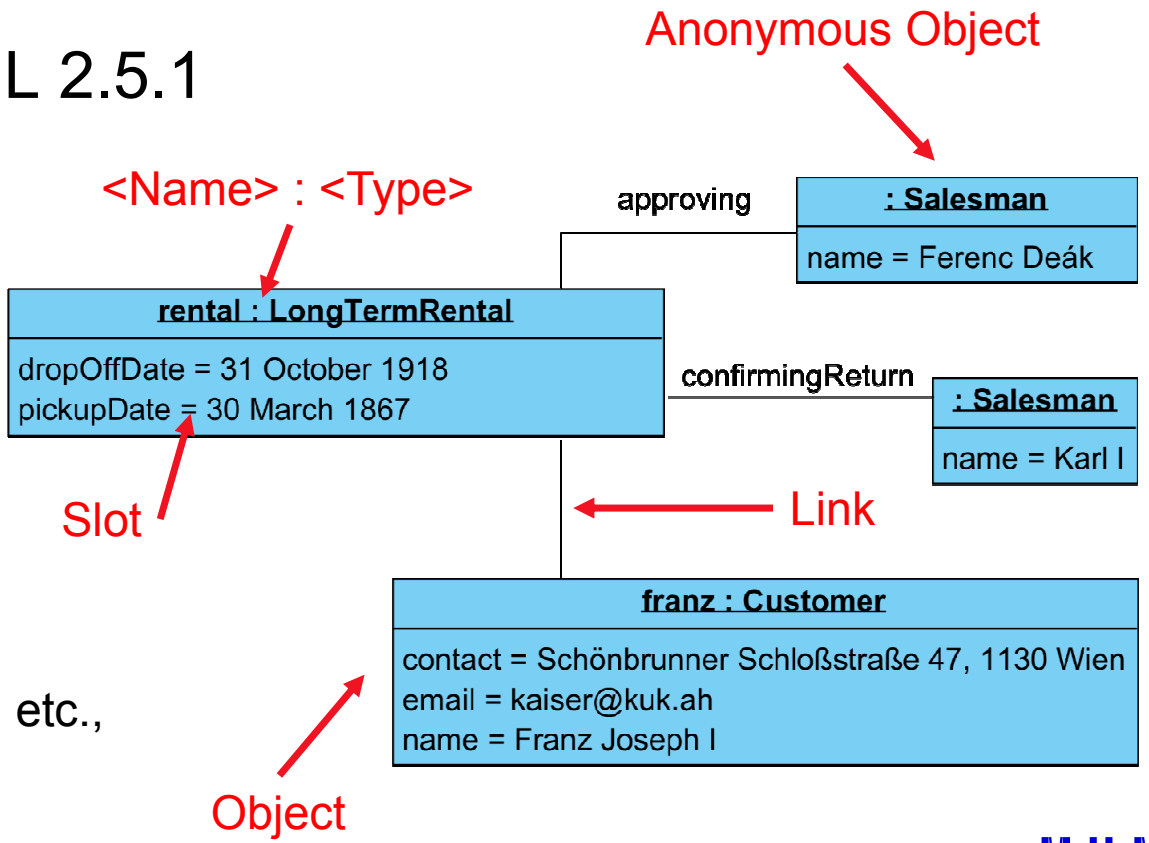
MUNI FI

Object Diagram – Why?

- Verify the correctness of a Class Diagram
  - Especially for some complicated case
- Illustrate and highlight some facts
  - Clarify complex relationships in Class Diagram
  - Understand behavior in a specific moment
- Essentially it is “hitting a breakpoint” in debugger
  - But in a visual form

Object Diagram – Components

- Rather underdefined in UML 2.5.1
- Objects
  - Instances of Classes
  - Optional name
  - Mandatory type
  - Fields with values = slots
- Links
  - Instances of Associations
  - Sometimes you can see multiplicity, etc., but it is not standard



MUNI FI

Task for this week

You gotta do what you gotta do

- Process the feedback
- Based on the Design Class Diagram, think about its interesting and complicated instantiations
  - Something you want to emphasize
  - Use Case Diagram and Activity Diagram should help you in finding it
- **Choose two cases and model them into two Object Diagrams**
  - If you need to change something in Design Class Diagram, just do it, no worries.
  - Just make sure that the changes are consistent across your project
  - **Write short description explaining which situation they represent**